A Formal Framework for Secure Routing Protocols

Chen Chen[1] Limin Jia[2] Hao Xu[1] Cheng Luo[1]
Wenchao Zhou[3] Boon Thau Loo[1]

[1] University of Pennsylvania, Philadelphia, PA 19104, USA
chenche, haoxu, boonloo@cis.upenn.edu
[2] Carnegie Mellon University, Pittsburgh, PA 15213, USA liminjia@cmu.edu
[3] Georgetown University, Washington, DC 20057, USA wzhou@cs.georgetown.edu

Border Gateway Protocol (BGP), the de facto Internet routing protocol, is vulnerable to various attacks. Redesigns of Internet routing infrastructure (e.g. S-BGP and SCION) have been proposed to address the security concerns with BGP. However, none of them formally verifies its security claims. Existing model-checking-based protocol analysis tools cannot be directly applied to verifying routing protocols, as it requires verification on an infinite number of network topologies. Proving small model theorems enables sound and complete verification on a finite number of topologies [**?**]; however, such theorems are specific to each protocol and may not exist for some protocols.

We develop a unified formal framework that combines development, empirical evaluation, and formal verification of secure routing protocols. The framework uses SeNDLog, a declarative networking language resembling distributed Datalog, as the protocol specification language. The specification can both be translated to executable code for performance evaluation and be used to generate proof obligations for verification.

We develop trace-based operational semantics for SeNDLog. The semantics adopts a distributed execution model. Network states are modeled as relational databases maintained at each node. Evaluation of SeNDLog programs dictates how nodes communicate with each other and how tuples in databases are updated incrementally.

Inspired by prior work on analyzing safety properties of programs executing concurrently with adversaries [1], we develop a sound Hoare-style program logic for SeNDLog. After specifying security properties in first-order logic, we use our program logic to derive invariant properties of the SeNDLog program by checking that each rule in that program maintains those invariant properties.

We implement a compiler for SeNDLog, and a verification condition generator (VCG) that extracts lemmas necessary to verify a given SeNDLog program. Verifying a secure routing protocol within our framework involves (1) encoding the protocol in SeNDLog; (2) specifying the security properties of the protocol and auxiliary properties of the program; (4) using VCG to generate proof obligations in a theorem prover (e.g. Coq); and (5) discharging the proof obligations. We encode S-BGP and SCION in SeNDLog and verify path authenticity properties of both protocols.

Keywords: declarative languages, secure routing protocols, verification.

References

1. Datta, A., Derek, A., Mitchell, J.C., Roy, A.: Protocol Composition Logic (PCL). Electronic Notes in Theoretical Computer Science 172, 311–358 (2007)